Journal of Complex Networks Advance Access published March 9, 2015

Journal of Complex Networks (2015) Page 1 of 14
doi:10.1093/comnet/cnv002

Opening bottlenecks on weighted networks by local adaptation to cascade failures

Jeff Alstott

Section on Critical Brain Dynamics, National Institute of Mental Health, Bethesda, MD, USA and Brain Mapping Unit, Behavioural and Clinical Neuroscience Institute, University of Cambridge, Cambridge, UK

Sinisa Pajevic

Mathematical and Statistical Computing Laboratory, Division of Computational Bioscience, Center for Information Technology, National Institutes of Health, Bethesda, MD, USA

Ed Bullmore

Brain Mapping Unit, Behavioural and Clinical Neuroscience Institute, University of Cambridge, Cambridge, UK

and

Dietmar Plenz†

Section on Critical Brain Dynamics, National Institute of Mental Health, Bethesda, MD, USA
†Corresponding author. Email:plenzd@mail.nih.gov

Edited by: Ernesto Estrada

[Received on 22 October 2014; accepted on 8 January 2015]

Structure and dynamics of complex systems are often described using weighted networks in which the position, weight and direction of links quantify how activity propagates between system elements, or nodes. Nodes with only few outgoing links of low weight have low out-strength and thus form bottlenecks that hinder propagation. It is currently not well understood how systems can overcome limits imposed by such bottlenecks. Here, we simulate activity cascades on weighted networks and show that, for any cascade length, activity initially propagates towards high out-strength nodes before terminating in low out-strength bottlenecks. Increasing the weights of links that are active early in the cascade further enhances already strong pathways, but worsens the bottlenecks thereby limiting accessibility to other pathways in the network. In contrast, strengthening only links that propagated the activity just prior to cascade termination, i.e. links that point into bottlenecks, eventually removes these bottlenecks and increases the accessibility of all paths on the network. This local adaptation rule simply relies on the relative timing to a global failure signal and allows systems to overcome engrained structure to adapt to new challenges.

Keywords: networks; bottleneck; branching process; activity cascade; learning; adaptation; synaptic plasticity.

1. Introduction

Many complex systems can be accurately described as networks in which the nodes, i.e. system components, interact via specific links [1,2]. The functioning of these systems is captured by the flow of activity or information between nodes along particular paths [3]. These paths may include many nodes, but tend to secure reliable and optimal functioning of the system. For example, highways link cities, phone lines connect individuals, and in the brain, white matter fibre tracts integrate distant cortical regions [4,5]. In these networks, high flow along specific pathways is realized by links with strong weights, whereas alternative routes are often characterized by links with low weights that are comparatively less efficient in transferring activity. Such an organization of preferred and non-preferred paths usually ensures activity goes where it is needed and does not go where it is not. The challenge for a particular network configuration, however, arises from the requirement of many real-world networks to adapt to changing needs and environments. Changes in commuting patterns, alterations in communication habits or learning a new skill will all require restructuring of how activity or information propagates in the system. Alternative pathways must be explored and strengthened while existing ones may need to be reduced in importance.

Such a reorganization of links in a network is rather demanding. For example, adaptation rules can strengthen particular pathways for preferential activity propagation [6]. These preferred pathways might make performing the ‘required’ system function, e.g. connecting specific sites in the network, easier; however, they also make learning future paths harder. Any activity heading out of preferred pathways runs into bottlenecks of weaker connectivity, preventing exploration of the paths beyond. One way for a system’s activity to explore weak, non-preferred pathways is to identify and open those bottlenecks as needed. With bottlenecks widened, the system’s activity and information flow can explore alternative pathways, which may subsequently be learned and strengthened.

Methods for identifying and widening bottlenecks have been developed in the context of communication networks and other engineering tasks [7–11]. These methods minimize bottlenecks by adopting an external, global point of view with complete knowledge and control of the network. Here, we introduce an adaptation process that does not rely on detailed global knowledge, yet allows for a system to identify and iteratively widen bottlenecks from the unfolding of its own internal dynamics. Many systems, such as social, communication or brain networks, feature system dynamics typical of cascades of activity propagating from node to node [12–17]. We show here that with information contained within endogenous cascading activity and a simple signal that a cascade has failed, it is possible to identify and widen bottlenecks, thereby increasing flow across all possible paths. In a first step, we identify a typical temporal profile of cascades and utilize it to develop an appropriate adaptation process, in which link weights are altered in response to endogenous network activity. We then show that a particular adaptation process is capable of widening bottlenecks and increasing flow across the network, allowing new paths to be explored.

2. Results

2.1 *The temporal profile of cascading activity*

We first identified the type of information one can extract by observing the spread of cascades on a network. We approached this problem using a Susceptible-Activated-Removed (SAR) model [18]. In this model, an activity cascade started on the network at a randomly selected node, then propagated along links to one or more connected nodes with a probability proportional to the weight of the connecting link. The cascade continued iteratively through subsequent generations of activated nodes,

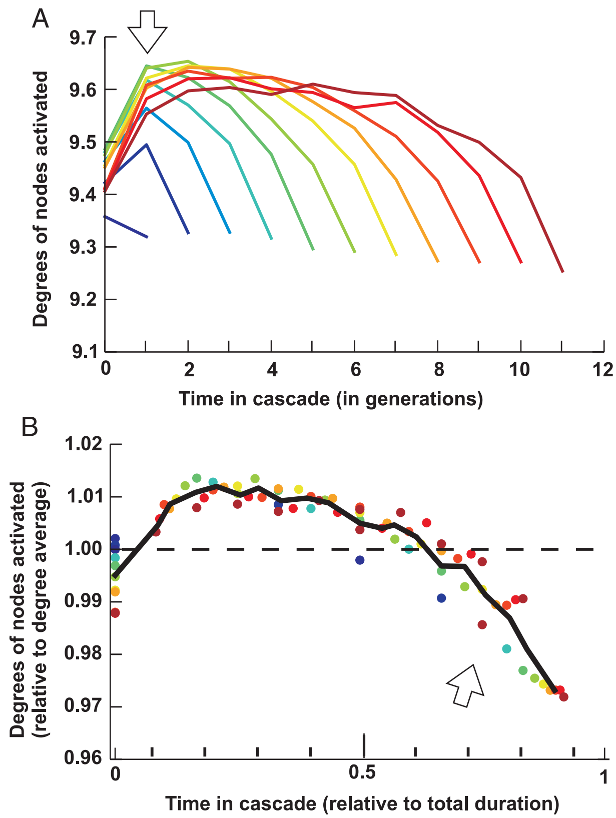

FIG. 1. Cascades showed a consistent temporal profile in the degrees of nodes visited during their formation. (A) The average node degree activated at each generation. Cascades tend to initially engage higher degree nodes and terminate at relatively lower degree nodes. *Dark Blue*: shortest cascades. *Dark Red*: longest cascades. (B) As in A, but rescaled to cascade duration and average degree of the nodes activated in the cascade. *Black line*: average over rescaled cascade lengths (20 bins). In the adaptation models, early stage adaptation increased the second link activated in the cascade (*A*, arrow), and last stage adaptation increased the last link activated in the cascade (*B*, arrow). 10^6 cascades were simulated, and are grouped here by the number of generations they had (2–12).

and terminated when none of the activated node's links successfully propagated. Nodes could only be activated once per cascade. We ran 10^6 cascades on Newman–Watts (NW) networks with either equal or random link weights (see Methods). The temporal profile of cascade behaviour was well conserved across varying cascade lengths (Fig. 1). Cascades that consistently activated higher degree nodes had longer durations (Fig. 1(A)). When normalized for cascade duration and the average degree of activated nodes in the cascade, we obtained a universal profile of cascade formation relative to node degree (Fig. 1(B)). Thus, early in a cascade, activity moved to higher degree nodes, as these nodes had more links by which activity could reach them. Upon reaching lower degree nodes, however, a cascade was more likely to terminate, as these nodes had fewer links with which to propagate activity onward. We defined the termination power of a node as the probability it would terminate a cascade when activated, i.e. number of terminations divided by number of activations. Indeed, termination power and node degree were inversely correlated ($r = -0.93, p < 0.001$; Student's t-test), demonstrating that low degree nodes were bottlenecks for cascades.

It had previously been suggested with this cascading model that terminations were more likely at highly clustered nodes [19]. In most networks, though, node degree inversely correlates with the clustering of a node. To separate the potential effects of node degree and node clustering on the termination power of a node, we created random configuration model networks in which all nodes had the same degree but otherwise random wiring, which allowed clustering to vary across nodes even though degree was constant (see Methods). On these networks, node termination power did not correlate with node clustering ($p > 0.1$; Student's t), thus concluding that the low node degree is the dominant feature causing terminations.

2.2 *Adapting to cascading activity*

Our initial exploration of cascade profiles was performed on networks with equal or random weights. In these networks, the node degree was found to determine bottlenecks, since it is proportional to the out-strength of a node. With network topology fixed, node strength can be modified by manipulating link weights, which may change bottleneck dynamics. We therefore introduced a weight adaptation process to the SAR model that made use of the insights gained from the cascade profile in Fig. 1. In this adaptation process, the network topology was fixed, but link weights were modified after a cascade terminated. Based on the temporal profile in Fig. 1(A), we identified two generations, i.e. link positions, which might be most effective in manipulating cascade dynamics. In early stage adaptation, the second link activated was selected (between generation 1 and 2; arrow in Fig. 1(A)). This link was likely to be between higher degree nodes. In last stage adaptation, the last link activated was increased in weight (Fig. 1(B); arrow). This link was likely to be entering a low degree node. After either adaptation, weights were converted to propagation probabilities by a scaling factor such that across all nodes the average out-strength remained close to one, i.e. dynamics remained approximately critical [19,20]. We ran this model for $7 * 10^6$ cascades on 100 directed NW networks with initially equal weight on all links.

Early stage adaptation created a rich-get-richer scenario. Activity went to high degree nodes early in the cascade, which increased the weight of the links attached to high degree nodes, which then made activity more likely to go there in subsequent cascades. The in- as well as out-strength of high degree nodes increased more and more, while the opposite was true for low degree nodes (Fig. 2(A,B)). This created a two-tiered structure of 'the rich and the rest', where high degree nodes were rarely the sites of cascade terminations (Fig. 2(C), dark regions), while low degree nodes made up a large portion of cascade terminations (Fig. 2(C), bright regions). This process continued rapidly for $\sim 10^6$ cascades, when the links on the highest degree nodes approached the maximum weight allowed in the model (Fig. 2(A,B), arrow; see Methods). Eventually, all links in the network increased towards the maximum weight, which slowly returned the network to the initial state of propagation strength being proportional to the corresponding out-degree of a node. The two-tiered bottlenecking structure highlighted by the early stage adaptation nevertheless persisted well into the saturation process (Fig. 2(C)).

Last stage adaptation, in contrast, targeted and widened bottlenecks, thereby reducing the strength of high degree nodes and increasing the strength of low degree nodes (Fig. 2(D,E)). This process started at the lowest degree nodes and iteratively worked its way towards the highest degree nodes (Fig. 2(F)). The iteration is detailed in Fig. 3, in which an active node is shown to target two nodes of different out-degree (Fig. 3; top), which translates into a higher failure rate for the lower degree node. Last stage adaptation increases the weights of the very links entering into this bottleneck (Fig. 3; lower right, 'adaptation'), which increases the in-strength of the lower degree node (Fig. 2(D), blue line), leading future activity to preferentially enter into this node and to potentially terminate. This, in turn, feeds into ever more in-strength increases and terminations (Fig. 3, 'Loop 1'). The increased activity at the

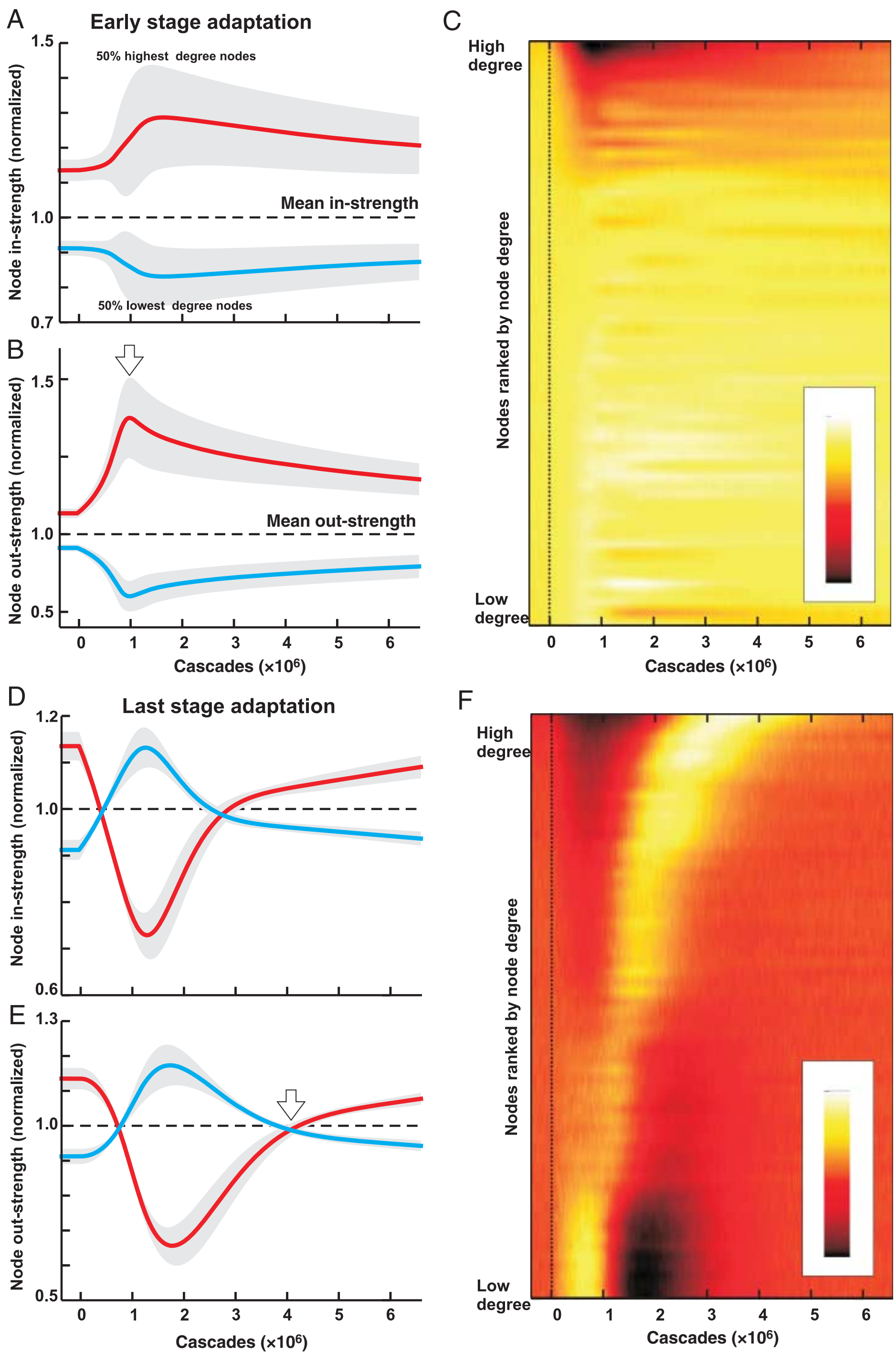
A
Early stage adaptation
50% highest degree nodes
Mean in-strength
50% lowest degree nodes
Node in-strength (normalized)
1.5
1.0
0.7
B
Node out-strength (normalized)
Mean out-strength
1.5
1.0
0.5
0 1 2 3 4 5 6
Cascades (×10⁶)
C
High degree
Low degree
Nodes ranked by node degree
Cascades (×10⁶)
D
Last stage adaptation
Node in-strength (normalized)
1.2
1.0
0.6
E
Node out-strength (normalized)
1.3
1.0
0.5
Cascades (×10⁶)
F
High degree
Low degree
Nodes ranked by node degree
Cascades (×10⁶)

FIG. 2. Early stage adaptation strengthens high degree nodes and worsens bottlenecks, while last stage adaptation iteratively targets and widened bottlenecks. Node properties in directed Newman–Watts networks of 60 nodes during early stage adaptation (A–C) and last stage adaptation (D–F). Across all figures, adaptation started after 10^6 cascades (vertical dotted lines). All values are means across 100 networks, and shading on line plots are standard deviations. (A–B) Node in-strength (A) and out-strength (B), grouped by high degree (red) and low degree (blue). All links started at the same weight, so strength initially directly corresponded with degree. During early stage adaptation high degree nodes grew even stronger, at the expense of the lowest degree nodes. Arrows denote the 'target' point of the adaptation process, which is returned to in Figs 4–5. (C) Heat map of where terminations occurred in the network across time. Vertical axis reflects nodes, ordered from low degree (bottom) to high degree (top). High degree nodes rarely terminated cascades (dark regions), while low degree nodes predominantly terminated activity (bright regions). (D–F) As A–C, for last stage adaptation. Low degree nodes gained in-strength (D), then out-strength (E), at the expense of high degree nodes. As low degree nodes gained in-strength terminations increased at these nodes (F). The little activity escaping through these bottlenecks then terminated at the next lowest degree nodes, increasing the out-strength of the initial bottlenecks and driving activity into the new bottlenecks. This process continued iteratively, shifting the bottleneck behaviour through ever-higher degree nodes until termination rates were roughly equal across all nodes.

low degree node, in addition, provides more opportunities to propagate activity along its few, weak, outbound links (Fig. 3; lower left, 'propagation'). The outbound links leading directly to relatively lower degree nodes are most likely to result in cascade termination, starting the cycle over again (Fig. 3, 'Loop 2'). As this next generation of links increased through last stage adaptation, it increases the out-strength of the original nodes, until they are no longer bottlenecks (Fig. 2(E), blue line; Fig. 3, 'Loop 2'). This process of driving activity into bottlenecks until they pass on enough activity to open up repeats itself through sets of nodes with ever-higher degrees (cf. Fig. 2(F)).

Because the adaptation process keeps the average propagation strength in the network constant, selective strengthening of low degree nodes weakens high degree nodes through such global normalization (Fig. 2(D,E), red lines). Eventually, high degree nodes become bottlenecks, which nevertheless are subsequently targeted and widened through last stage adaptation. This is visible in Fig. 2(F) by the zone of high cascade termination that moves through the network towards higher degree notes until it vanishes after $\sim 4 * 10^6$ cascades and all nodes exhibit similar termination power. The bottleneck widening process therefore adjusts link weights such that it compensates for low degree with a weight increase, severing the initial simple relationship between degree and strength.

Accordingly, early stage adaptation increased the variation or disparity of out-strength on the network, with high degree nodes gaining even larger strength and low degree nodes having even less strength. This out-strength disparity increased by $\sim$1,000%, until link weights started to saturate (Fig. 4(A)). In last stage adaptation, on the other hand, out-strength disparity fluctuated at different points in the iterative bottleneck widening process, ending in an extended period of decreased out-strength disparity (Fig. 4(B)). During this period out-strength disparity was decreased $\sim$60%, with all nodes having roughly equal strength. We identify these two regimes as the target points of the two adaptation processes (Fig. 4(A,C); arrows; cf. Fig. 2). At these target points, the temporal profile of cascades had changed (Fig. 4(C,D)). Early stage adaptation made cascades more likely to terminate in low degree nodes (Fig. 4(C), right arrow). Last stage adaptation, in contrast, flattened the temporal profile and cascades were less likely to terminate at a low degree node (Fig. 4(D), right arrow). The early portions of cascades similarly had lowered preference for high degree nodes after last stage adaptation (Fig. 4(D), left arrow). After early stage adaptation the first generation of cascades was also more likely to be at high degree nodes (Fig. 4(C), left arrow), as lower degree starting nodes had such low out-strength that they were less likely to propagate activity to any subsequent generation. These differential changes in out-strength for networks were also found if links were undirected, were initialized with heterogeneous weights or when a random, Erdős–Rényi topology instead of small-world topology was used (Fig. 5(A); for details see Methods).

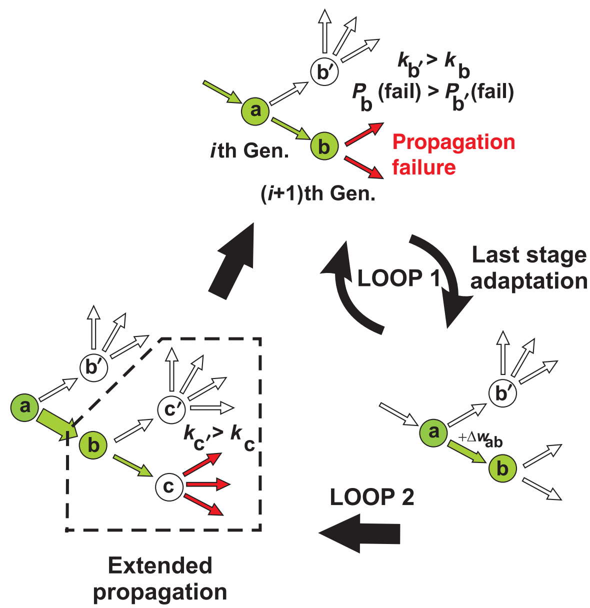

FIG. 3. Last stage adaptation opened bottlenecks through an iterative adaptation process. Top: Propagating activity can enter high degree nodes or low degree nodes. Activity entering low degree nodes has a higher probability of failing to propagate activity, as these nodes have fewer links. These low degree nodes are the bottlenecks to activity propagation. Bottom right, Adaptation: When a bottleneck node terminates activity, last stage adaptation increases the weight of the link going into that node. This increases the probability that activity will enter into the bottleneck in the future, repeating the process (Loop 1). Bottom left, Propagation: As more activity reaches the bottleneck, some fraction of the activity will propagate outward. This activity then goes through the same process as the previous generation (Loop 2). This second generation of adaptation increases the out-strength of the bottleneck node, thus widening the bottleneck. k_b: out-degree of node b; *i*th Gen.: *i*th generation; $P_b(fail)$: Probability of node b to not propagate. $+\delta W_{ab}$: increase in weight for link a to b. Broken line: network element of Loop 2 that re-enters Loop 1 for successive modifications. Note that out-degree $k_b > k_c$ and $k_{c'} > k_{b'}$ to indicate the iteration through higher and higher out-degrees in the network.

The different bottleneck states reached by the two adaptation processes reflected themselves in the structure of pathways in the network. We measured the positioning and accessibility of nodes via multiple possible weighted paths using eigenvector centrality [21]. The average centrality of all nodes decreased with early stage adaptation, but increased with last stage adaptation (Fig. 5(B)). These changes were particularly notable among the least central nodes in the network; the minimum centrality across all nodes became lower after early stage adaptation, but increased with last stage adaptation (Fig. 5(B)). The changes in centrality reflected the accessibility of nodes via multiple possible paths along many possible nodes. The best-case, shortest paths between two nodes, however, changed differently in response to the adaptation processes. The average shortest weighted path length between any two nodes on the network increased after both early stage adaptation and last stage adaptation, though less after last stage adaptation (Fig. 5(B)). The weight changes changed the clustering and community structure of the network. Local weighted clustering decreased with early stage adaptation, but increased with last stage adaptation (Fig. 5(C)). The same alterations to community structure appeared at a global scale, with early stage adaptation lowering the modularity structure of the entire network, while last stage adaptation increased it (Fig. 5(C)). The decreased path length and increased clustering created by the last stage adaptation made the network more of a small world [22], while the early stage adaptation made it less of one.

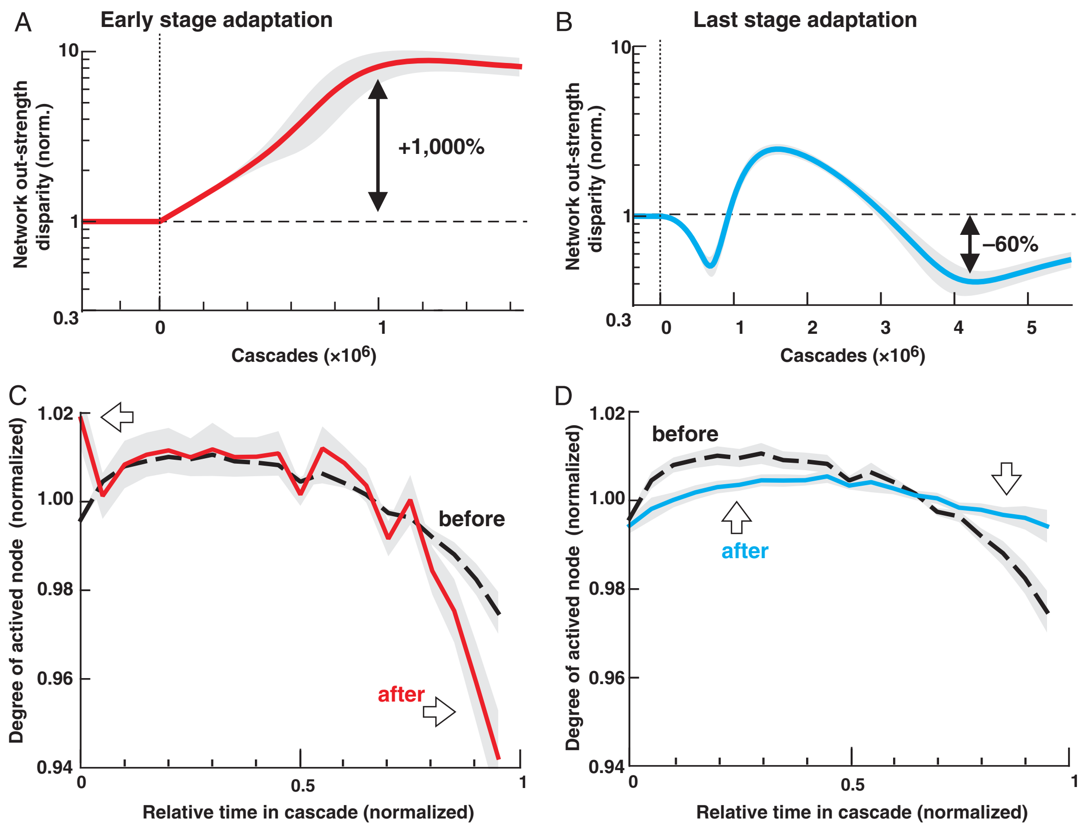

FIG. 4. Network out-strength disparity increased with early stage adaptation and decreased with last stage adaptation, respectively, increasing and decreasing the severity of bottlenecks. (A and B) Network out-strength disparity (coefficient of variation) during adaptation for early stage adaptation (A, red) and last stage adaptation (B, blue). Values are expressed relative to the initial values before adaptation began, and are means across 100 networks, with standard deviations represented by shading. Arrows indicate the target point, which for early stage adaptation is the weight saturation point and for last stage adaptation is immediately after the iterative bottleneck opening process. Early stage adaptation increased the out-strength disparity by ~1,000%, while last stage adaptation decreased the disparity by 60%. (C–D) The typical temporal profile of cascades, relative to their total number of generations and the average degree of the nodes they activated (As Fig. 1(D)). Lines are averages over the mean cascade profile for cascades of lengths 2 through 12, with 20 bins. Black lines: The initial profile before adaptation. Coloured lines: The profile after adaptation, at the target point. (C) After early stage adaptation, the topological profile of cascades was made more severe. Cascades were more likely to end at low degree nodes (right arrow). The first generation of cascades was also more likely to initiate at high degree nodes (left arrow), as lower degree starting nodes had such low out-strength that they were less likely to propagate activity to any subsequent generations. (D) After last stage adaptation, the temporal profile of cascades was flattened. Cascades initially visited nodes with degrees only slightly above the degree average (left arrow) and terminated at nodes with degrees only slightly below the average (right arrow).

The weight changes introduced by the adaptation mechanisms also changed the amount of activity in cascades. The number of activations in cascades grew with early stage adaptation, as the prominent high degree nodes increased the opportunity for cascades to branch (Fig. 5(C)). Last stage adaptation, on the other hand, shrank cascade sizes, as the branching capabilities of the high degree nodes were depressed.

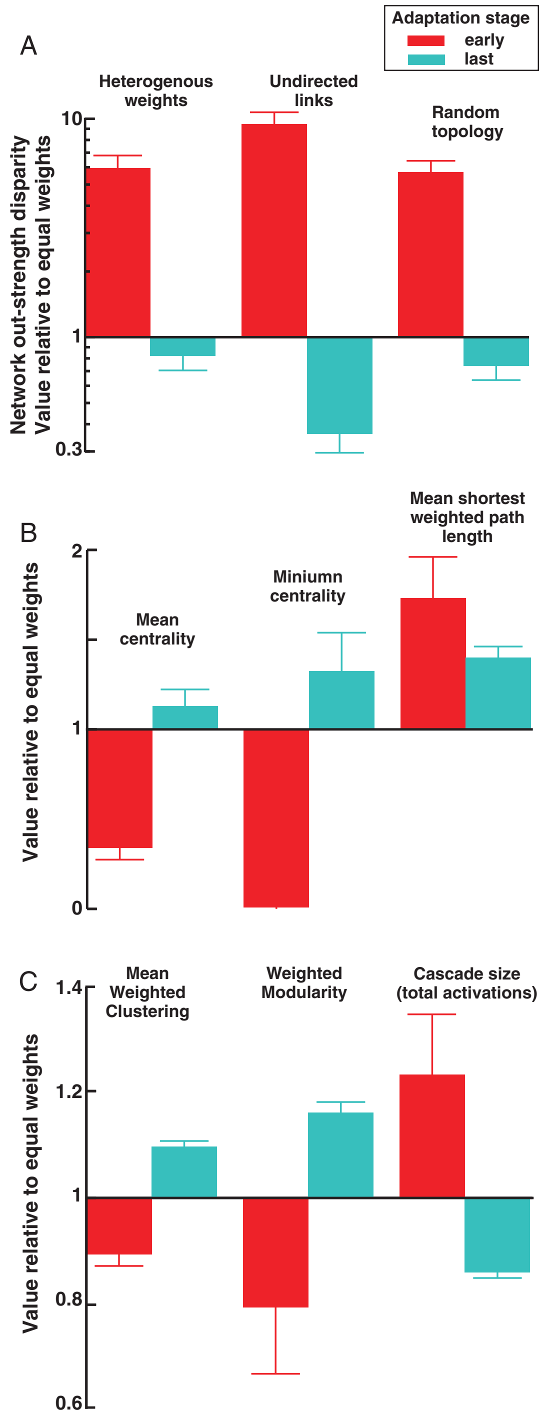
Adaptation stage
early
last
A
Heterogenous weights
Undirected links
Random topology
Network out-strength disparity
Value relative to equal weights
10
1
0.3
B
Mean shortest weighted path length
Miniumn centrality
Mean centrality
Value relative to equal weights
2
1
0
C
Mean Weighted Clustering
Weighted Modularity
Cascade size (total activations)
Value relative to equal weights
1.4
1.2
1
0.8
0.6

Fig. 5. Early and last stage adaptation were consistently different on multiple network types, and yielded different functional effects. Measures at the target point for early stage adaptation (red) and last stage adaptation (blue). All values are expressed relative to the initial values before adaptation began. All values are means across 100 networks, with standard deviations represented by error bars. (A) Out-strength disparity for different types of networks. Whereas results in other figures showed the behaviour of directed Newman–Watts networks starting from homogenous weights before adaptation, none of these network properties are required to produce the changes in out-strength disparity. Left: Undirected NW networks, with equal starting weights on all links. Center: Directed NW networks, with heterogeneous link starting weights (uniform distribution between 0.1 and 1; see Methods). Right: Directed Erdős–Rényi networks, with equal starting weights on all links. (B and C) Measures of the network structure and activity at the target point, on directed NW networks. (B) Last stage adaptation improved nodes accessibility via many paths at the expense of the shortest paths and total network activity. Early stage adaptation lowered nodes' accessibility as well as the shortest paths, and increased network activity. Measures: mean node eigenvector centrality across the network, minimum node eigenvector centrality on the network and mean weighted shortest path length between every pair of nodes in the network. (C) Last stage adaptation increased local and global community structure, while decreasing the average size of cascades. Early stage adaptation decreased local and global community structure, while increasing the total size of cascades. Measures: average weighted local clustering, weighted modularity of the entire network and mean number of activations per cascade (calculated from a sample of 20,000 cascades centered at the target point).

As further controls, we tested the specificity of the two adaptation mechanisms by comparing their behaviour to other link positions. Second-to-last stage adaptation showed a transitory period in network reconfiguration after which the initial condition was reestablished without an improvement or change in out-strength disparity (1.0 ± 0.03), as expected from the cascade temporal profile in Fig. 1. Similarly, adapting those links emanating from the initial, randomly chosen nodes in the network only marginally increased out-strength disparity (1.33 ± 0.12 at target point).

3. Discussion

Here, we have shown how networks can reorganize themselves to improve network flow using a local adaptation rule operating on bottlenecks identified by cascade failures. The crucial insight into the adaptation process was gained on networks with initially random or constant link weights by demonstrating that activity flow initially moves to high degree hubs early in cascades and terminates at low degree bottlenecks at the end of cascades. Accordingly, adapting the weights of links that are active early vs. last in the cascade has very different consequences. Early stage adaptation preferentially strengthens hubs and worsens bottlenecks at low degree nodes. The reverse is true for last stage adaptation, which opens low degree bottlenecks and iteratively equalizes out-strength inhomogeneities throughout the network. Functionally, bottlenecks to network flow or traffic could prevent a network from adapting to changing demands, e.g. routing traffic through different pathways. When the requirement arises to route traffic through new pathways and cascades exhibit high failure rates at bottlenecks, our work demonstrates that the activity-driven last stage adaptation opens these bottlenecks by directing activity into them. This initially non-intuitive direction in adapting to failure, in fact, not only opens bottlenecks, but also raises the centrality of formerly isolated nodes. These changes should increase the ability for activity to move from anywhere in the network to anywhere else, via any arbitrary route. Thus, the state created by last stage adaption is a 'smooth' one, from which a network could most easily learn new associations or carve new pathways.

3.1 *Strengthening hubs vs. bottlenecks*

Early stage adaptation strengthened the connectivity of hubs. Hubs can have considerable influence on system dynamics through such means as lowering the shortest paths between disparate network regions

[22]. However, the best possible paths are only relevant for certain system configurations [23]. For example, the packet switching and routing behaviour of telecommunication network directs activity along many pathways at once. For a brain producing a particular behaviour, the necessary activation pattern may be a long and circuitous path. Hubs do not improve the connectivity of the most weakly connected paths, which may not include the high connectivity hubs. Early stage adaptation and the hubs they strengthen improve best-case scenarios, but not worst-case scenarios. Last stage adaptation, in contrast, targets worst-case paths by adding weight to those areas that are not easily accessible. This results in a network state in which all nodes have more equalized ability to propagate activity.

Adapting into the equalized state may be useful in systems like communication networks and power grids, which must balance loads throughout the network [7,8]. On communication networks, the equalized state can also aid in combating denial of service attacks. This state may also be useful for adaptive networks that must learn new sequences of activity; raising the profile of worst-case routes through the network by increasing the likelihood that they will be explored and potentially identified for future learning, perhaps to turn into new preferred paths.

3.2 *Bottlenecks intensify before they weaken*

Last stage adaptation has the advantage of using local information only from those portions of the network that have actually received activity. This feature makes congestion worse before it gets better, as demonstrated in the examples of building road works. If last stage adaptation were employed in road works, the signal to adapt would arrive when traffic backs up on streets (links) that lead into an intersection with few good exits (a low out-strength node). The slowing or stopping of traffic equates to the failure signal leading to a widening of the streets on which the backups occur. This initially makes the situation worse, as more cars arrive on the wider road and the cause for the backup, which is actually the lack of good exits, i.e. low node out-strength, is not removed. The solution arrives by more and more traffic exploring the existing, but weak exits at the intersection, such as an alley. If this alley also leads to a place with few good exits, it will also backup. Once there is sufficient backup on the alley, the alley will widen into a road, and so the process will repeat itself. This process at first seems circuitous, however, the alternative solution, which is widening the existing weak exits in the first place, faces two problems. First, it requires an *a priori* identification of weak exists and selection of which one to widen, and secondly, it only moves the traffic backup problem by 1 step, as the widened exit might point into the next intersection with weak exits. This iteration employed by last stage adaptation can open many bottlenecks, but it will not work well if a weak exit establishes an on-ramp to a highway. In that case, any traffic using the weak exit will leave without backing up, and no adaptation signal to widen the alley is generated. Thus, last stage adaptation may be most appropriate for exploring vacant portions of the network where there are no ‘highways’.

3.3 *Cascade-level adaptation in the brain*

Cascade-level adaptation mechanisms are especially relevant for learning algorithms in the brain. In the mammalian cortex, one of the most important learning rules identified is spike-timing-dependent plasticity [24–26]. In STDP, the strength of a synaptic connection from neuron a to b, i.e. link weight from node a to b, will be strengthened if a spike in a initiates/arrives before a spike in neuron b, i.e. node a is active before node b. STPD governs directional, i.e. causal, relationships. As STDP is a node-level adaptation rule, it is usually not explored at the cascade-level. However, early and last stage adaptation demonstrate how STDP can have different effects at the system level when applied at different points in a cascade. As detailed using the schematics in Fig. 3, if neuron b fires a spike, but does not recruit postsynaptic neurons c or c', activity propagation will fail. With last stage adaptation, this will lead to strengthening of the connection from a to b in line with both neurons having fired in the right temporal order and close to the termination of the cascade. As detailed above, this will iteratively engage 'Loop 1' and 'Loop 2' to further enhance synaptic links between neuron a and b. This will eventually increase the ability of neuron b to recruit postsynaptic neuron c and will extend activity propagation. We point out the *a priori* selection alternative, discussed in the traffic example above, would be particularly challenging in the brain, as each neuron projects to $\sim 10^4$ neurons. Our work shows that applying STDP in early or last stage adaptation will have a very different outcome in the reorganization of a neuronal network. Importantly, the last stage adaptation algorithm has neurophysiological underpinnings. A rich body of literature links learning in the brain to a global prediction error signal physically realized by a transient release of specific neuromodulators [27,28]. Such prediction error signals are in line with the cascade failure signal used in our study. Specifically, if STDP is only employed at synapses that were active immediately before a prediction error signal, then last stage learning is in line with the well-established temporal difference rule in reinforcement learning [29]. In this rule, connections along a neuronal pathway are strengthened proportional to how close they have been activated in time with respect to an achieved or missed goal. If indeed reinforcement learning in the brain acts similarly compared with our last stage adaptation process, rewarding the endpoint of a particular action or encouraging repeat attempts might provide the seed of adaptability to changing environmental conditions.

3.4 *Summary and conclusion*

Adaptive systems must be able to create new behaviours, and bottlenecks to activity propagation could be a barrier to that end. By adapting to ongoing cascades of activity on a network, it is possible to widen or remove the bottlenecks and improve access to a more diverse set of possible network paths. Mechanisms such as last stage adaptation may allow systems to remain adaptive and flexible, and thus improve their functioning in response to changing needs and environments.

4. Methods

4.1 *Network topologies and weight structures*

Model networks were created with two topologies, each with 60 nodes and an average degree of approximately 10: Newman–Watts ($p = 0.033$, $K = 4$, [30]) and Erdős–Rényi ($p = 0.166$, [31]). Model networks in which all nodes had the same degree but varied clustering were also created, by assigning each node the same number of links but otherwise wiring randomly ($N = 500$, $k = 10$). Both directed and undirected versions of networks were created. In directed networks, all links were reciprocal, such that if a link existed from node i to node j then a link also existed from node j to node i. Link weights were initially assigned to be either homogeneous (such that all weights were equal to 1) or heterogeneous (uniformly distributed between 0.1 and 1).

4.2 *Activity propagation*

Cascades of activity were simulated on the networks by a SAR model [18]. Activity began at a random node and probabilistically propagated along links to other nodes. The probability of a link to propagate activity was proportional to the weight of the link. Weights were converted to probabilities by a scaling factor such that across all nodes the average strength (undirected networks) or out-strength (directed networks) was 1. If no links exiting a node propagated, then activity terminated at that node. An activated node could not be reactivated in a single cascade, and thus could not propagate activity for the remainder of that particular cascade. After activity ceased, the weights of links that were active in the cascade were updated according to an adaptation process (described below). All nodes were then reset to being susceptible to activation, and another cascade was initiated at a randomly selected node.

4.3 *Adaptation process*

After each activity cascade a specific link's weight, w_t was increased to a new weight w_{t+1}, according to an adaptation process proposed in [19]:

$$w_{t+1} = w_t \left(1 + p_0 \frac{w_{\max} - w_t}{w_{\max}}\right) \tag{1}$$

where $w_{\max}$ is the maximum possible weight and p_0 is a small incremental factor. We use here $w_{\max}$ equal to 5 and p_0 to 0.001. A range of other parameter values was explored, giving qualitatively similar results.

The specific link updated by the adaptation process was chosen by its position in the generations of the activity cascade. These were: the second link ('early stage adaptation'; the first link activated after the random starting node's links propagated) and the last link ('last stage adaptation'; the link pointing into the cascade's termination node). As controls, we also adapted links from which the randomly selecting starting node propagated activity only or second-to-last link terminations. For any link selection process, if the cascade branched and resulted in multiple links being simultaneously active in the selected generation of links, then all such links were updated.

Funding

This work was supported by the National Institutes of Health Intramural Research Program of the National Institute of Mental Health and the Division of Computational Bioscience/Center for Information Technology, the UK Medical Research Council (MC_US_A060_0046) and the Wellcome Trust. Ed Bullmore is employed half-time by the University of Cambridge, UK, and half-time by GlaxoSmith Kline (GSK).

References

1. Albert, R. & Barabási, A. L. (2002) Statistical mechanics of complex networks. *Rev. Modern Phys.*, **74**, 47–97.
2. Newman, M. E. J. (2003) The structure and function of complex networks. *SIAM Rev.*, **45**, 167–256.
3. Ahuja, R. K., Magnanti, T. L. & Orlin, J. B. (1993) *Network Flows: Theory, Algorithms, and Applications*. Upper Saddle River, NJ: Prentice Hall, 846 pp.
4. Amin, S. M. & Wollenberg, B. (2005) Toward a smart grid: power delivery for the 21st century. *IEEE Power Energy Mag.*, **3**, 34–41.
5. Mišić, B., Sporns, O. & McIntosh, A. R. (2014) Communication efficiency and congestion of signal traffic in large-scale brain networks. *PLoS Comput. Biol.*, **10**, e1003427.
6. Wills, T. J., Lever, C., Cacucci, F., Burgess, N. & O'Keefe, J. (2005) Attractor dynamics in the hippocampal representation of the local environment. *Science*, **308**, 873–6.

7. Asztalos, A., Sreenivasan, S., Szymanski, B. & Korniss, G. (2012) Distributed flow optimization and cascading effects in weighted complex networks. *Eur. Phys. J. B*, **85**, 288.
8. Korniss, G., Huang, R., Sreenivasan, S. & Szymanski, B. K. (2011) Optimizing synchronization, flow, and robustness in weighted complex networks. *Handbook of Optimization in Complex Networks: Communications and Social Networks* (M. T. Thai & Panos M. Pardalos eds), vol. 58. Berlin: Springer, pp. 61–96.
9. Tizghadam, A. & Leon-Garcia, A. (2010) Betweenness centrality and resistance distance in communication networks. *IEEE Netw.*, **24**, 10–16.
10. Wu, Z. X., Peng, G., Wang, W. X., Chan, S. & Wong, E. W. M. (2008) Cascading failure spreading on weighted heterogeneous networks. *J. Stat. Mech. Theory Exp.*, **2008**, P05013.
11. Yang, R., Wang, W. X., Lai, Y. C. & Chen, G. (2009) Optimal weighting scheme for suppressing cascades and traffic congestion in complex networks. *Phys. Rev. E*, **79**, 026112.
12. Pickard, G., Pan, W., Rahwan, I., Cebrian, M., Crane, R., Madan, A. & Pentland, A. (2011) Time-critical social mobilization. *Science*, **334**, 509–12.
13. Cheng, J., Adamic, L. A., Dow, P. A., Kleinberg, J. & Leskovec, J. (2014) Can cascades be predicted? *Proceedings of the 23rd International Conference on World Wide Web*. Seoul: Republic of Korea. doi:10.1145/2566486.2567997. URL http://arxiv.org/abs/1403.4608.
14. Feldmann, A., Gilbert, A. C. & Willinger, W. (1998) Data networks as cascades. *Proceedings of the ACM SIGCOMM'98 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication—SIGCOMM'98*. New York, USA: ACM Press, pp. 42–55. doi: 10.1145/285237.285256. URL http://portal.acm.org/citation.cfm?doid=285237.285256.
15. Motter, A. & Lai, Y. C. (2002) Cascade-based attacks on complex networks. *Phys. Rev. E*, **66**, 065102.
16. Beggs, J. M. & Plenz, D. (2003) Neuronal avalanches in neocortical circuits. *J. Neurosci.*, **23**, 11167–11177.
17. Thiagarajan, T. C., Lebedev, M. A., Nicolelis, M. A. L. & Plenz, D. (2010) Coherence potentials: loss-less, all-or-none network events in the cortex. *PLoS Biol.*, **8**, e1000278.
18. Vespignani, A. (2011) Modelling dynamical processes in complex socio-technical systems. *Nat. Phys.*, **8**, 32–39.
19. Pajevic, S. & Plenz, D. (2012) The organization of strong links in complex networks. *Nat. Phys.*, **8**, 429–436.
20. Pajevic, S. & Plenz, D. (2009) Efficient network reconstruction from dynamical cascades identifies small-world topology of neuronal avalanches. *PLoS Comput. Biol.*, **5**, e1000271.
21. Borgatti, S. P. (2005) Centrality and network flow. *Soc. Netw.*, **27**, 55–71.
22. Cohen, R. & Havlin, S. (2003) Scale-free networks are ultrasmall. *Phys. Rev. Lett.*, **90**, 058701.
23. Estrada, E. & Hatano, N. (2008) Communicability in complex networks. *Phys. Rev. E*, **77**, 036111.
24. Markram, H., Lübke, J., Frotscher, M. & Sakmann, B. (1997) Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science*, **275**, 213–5.
25. Bi, G. Q. & Poo, M. M. (1998) Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neurosci.*, **18**, 10464–72.
26. Morrison, A., Diesmann, M. & Gerstner, W. (2008) Phenomenological models of synaptic plasticity based on spike timing. *Biol. Cybernet.*, **98**, 459–78.
27. Schultz, W. & Dickinson, A. (2000) Neuronal coding of prediction errors. *Ann. Rev. Neurosci.*, **23**, 473–500.
28. Potjans, W., Diesmann, M. & Morrison, A. (2011) An imperfect dopaminergic error signal can drive temporal-difference learning. *PLoS Comput. Biol.*, **7**, e1001133.
29. Sutton, R. & Barto, A. (1998) *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press.
30. Newman, M. E. & Watts, D. J. (1999) Scaling and percolation in the small-world network model. *Phys. Rev. E*, **60**, 7332–42.
31. Erdős, P. & Rényi, A. (1960) On the evolution of random graphs. *Math. Inst. Hungarian Acad. Sci.*, **5**, 17–61.